

SYSTEMS AND METHODS FOR ELECTRONIC INTERACTION

U.S. Patent Application

This application for U.S. patent is filed as an application for U.S. patent under U.S.C., Title 35, §111(a), and claims priority to U.S. provisional application, which was assigned Serial No. 60/170068 and filed on December 10, 1999.

Field of the Information

The present invention relates to novel systems and methods for public and/or private electronic interaction between industries for purposes of transmitting selective information to targeted individuals or entities within such industries, such as, between homebuilders and real estate professionals to facilitate home purchase transactions.

Background

In the homebuilding business, a homebuilder customarily pays a sales commissions to real estate brokers, agents and realtors in the event that clients, that they have referred, buy homes from or build homes through the homebuilder.

In one form of advertising, and from time to time, homebuilders engage in promotional campaigns by sending promotional materials to those members of the real estate industry. Because these materials are usually in print, homebuilders must pay for design, production, printing, mailing

lists rentals, postage and mailing services. These promotional campaigns can be very costly and are usually time intensive. For example, costs for each promotional campaign can range from \$400 to \$4000 and can take several weeks to produce and send. Notwithstanding the high costs and time involved, it is estimated that only about 10% of the industry members reached by these promotional campaigns are professionals who want to do business with homebuilders. Moreover, the information promoted by the homebuilders in their promotional campaigns is quickly dated due to, for example, fluctuations in costs to homebuilders and interest rates, which requires the promotional campaigns to be frequently repeated with updates.

In addition, a homebuilder generally requires that real estate professionals accompany each client on each initial visit to the homebuilder's sales office in order for the professionals to register themselves and their clients, so that commissions will be paid to the appropriate professionals in the event sales are made to their clients. Unfortunately, because this requirement is very time consuming to the real estate professionals, it generally acts as a deterrence and hindrance to home purchases, as opposed to an incentive for home purchases by clients of real estate professionals.

In summary, the problems with the current methods employed by homebuilders to do business with real estate professionals are costly, slow, inefficient in reaching targeted markets, and time consuming, they provide promotional information which becomes quickly dated, and they in many instances, inhibit real estate professionals from doing business with homebuilders.

Consequently, there is a need for a system and method for homebuilders to efficiently, quickly and inexpensively reach targeted markets with current promotional and marketing information and to induce and encourage all real estate professionals to register their clients with homebuilders, without the disadvantages discussed above.

Summary of the Invention

The present invention overcomes and alleviates the above-mentioned drawbacks and disadvantages of doing business between industries through the discovery of novel systems and methods which permit public or private, immediate electronic interaction and communication between industries.

Generally speaking, the present invention enables each industry to selectively interact and communicate electronically with one another on demand in public or private forums through the

internet without the costs and time disadvantages required heretofore. More specifically, the current invention allows each industry to provide current and complete information to one another instantaneously and to remain updated at all times, so that each industry benefits.

In the context of the real estate professional and homebuilder industries, the novel systems and methods of the present invention permit immediate electronic interaction between homebuilders and real estate professionals to facilitate home purchase transactions. Through the novel systems and methods of the current invention, homebuilders can continuously transmit current marketing and promotional information to real estate professionals to directly or indirectly induce them to encourage their clients to purchase homes from homebuilders. Also of importance, the novel systems and methods of the present invention afford real estate professionals the opportunity to electronically register their clients online with homebuilders of their choice without the necessity to resort to current methods or requiring *in persona* visits by the real estate professionals with their clients to homebuilders' sales offices and afford homebuilders with the opportunity to generally or specifically promote their inventory with complete and current information, without the necessity to resort to current methods of requiring homebuilders to engage in traditional promotional campaigns.

Accordingly, it should now be appreciated by those versed in this art that the novel systems and methods of the present invention eliminate the costs incurred by homebuilders for design, production, printing, mailing list rentals and postage associated with the traditional promotional campaigns. In addition, the novel systems and methods of the present invention permit the homebuilders to prepare and send the promotional information within minutes, thereby ensuring that all information remains current and reaches the targeted audiences, e.g., real estate professionals who want to do business with homebuilders, and to receive information from real estate professionals within minutes, then by reducing the time required for the homebuilders' sales staff to, for example, register those clients referred to them by the real estate professionals. The novel systems and methods of the present invention not only eliminate the necessity for real estate professionals to actually visit the sales offices of homebuilders with their clients for the purposes of registering their clients, they encourage real estate professionals who ordinarily do not do business with home builders to do so. Moreover, the novel systems and methods of the present invention permit real estate professionals to provide homebuilders with contact information on potential customers who they might otherwise not learn about.

The above features and advantages of the present invention will be better understood with reference to the following Detailed Description, Figs. and Example. It should also be understood that the particular systems and methods illustrating the present invention are exemplary only and not to be regarded as limitations of the present invention.

Brief Description of the Figs.

Reference is now made to the accompanying Figs. in which a more complete understanding of how the novel systems and methods of the present invention may be accomplished and practiced by reference to the following Detailed Description and Example, when considered together with the accompanying Figs.

Fig. 1 is a home web page from which a homebuilder and a real estate professional can learn about the program and click links to begin using it.

Fig. 2 is a web page that a homebuilder can obtain the logo link, so that he or she can add it to his or her web site in order to get more real estate professionals to participate in the program.

Fig. 3 is a web page that a homebuilder can bulk email to real estate professionals.

Fig. 4 is a web page that provides instruction for a homebuilder to bulk email a message to real estate professionals in a greeting card format.

Fig. 5 is a web page that allows a homebuilder to preview and mail his or her animated greeting card.

Fig. 6 is a web page that is an example of how an animated card looks and provides the geographic selection and sending mechanism.

Fig 7 is a web page that allows a real estate professional to locate communities where he or she can register his or her clients online.

Fig 8 is a web page that shows typical community listings.

Fig 9 is a registration web page that appears when a real estate professional selects a community listing, and when the real estate professional performs the actual online registration.

Fig 10 is a web page that confirms that the registration of Fig 9 was successful.

Fig 11 is an email generated by an online registration and is sent to a homebuilder with copies to the real estate professional and Brokers Welcome.

Fig 12 is a web page that invites real estate professionals to add themselves to the email mailing list selecting a free or enhanced listing.

Fig 13 is a web page that represents a free sign up page for real estate professionals to join the mailing list.

Fig 14 are web pages that allow real estate professionals to purchase an enhanced listing on realtorlist.net while adding themselves to the mailing list.

Fig 15 is a web page that appears after real estate professionals submit their information to the mailing list.

Detailed Description

By way of illustrating and providing a more complete appreciation of the present invention and many of the attendant advantages thereof, the following detailed description is given concerning the novel systems and methods.

Usage instructions for both homebuilders and real estate professionals are maintained at, for example, a web site, such as, <http://www.brokerswelcome.com>. From there, at least four transactions can occur.

First, real estate professionals can privately add their names and contact information to the mailing list. The real estate professionals can enter, for instance, name, company, physical address, region within their state, email address and phone number. This information is stored in a private, flat file database accessible by the homebuilders selected by the professional.

Second, real estate professionals can privately register their clients with selected homebuilders. To accomplish this, the real estate professionals can search for new home communities by city or county, price range and type of home. The results display descriptive listings of communities where the home builders have agreed to permit online client registration. From these listings, the real estate professional can click on to the homebuilders web site, in private, for more information. Should the real estate professional decide to register his or her client at the community, he or she clicks a registration link above the community listing. This brings him or her to the registration form where he or she enters contact information about himself or herself and the client he or she is registering. Upon clicking the "submit" button the information is emailed to the homebuilder of choice in private.

Third, homebuilders obtain, for example, the html coding, which is a logo link, such as, "Brokers Welcome" to be added to their web sites. This logo link is designed to induce real estate professionals visiting the homebuilders web sites to click them in order to receive web pages that encourage them to add themselves to the mailing list. The information entered is stored on the server in a flat file format in private.

Fourth, homebuilders can privately send emails to real estate professionals who have added themselves to the list. The homebuilders have at least two choices: A text message or an animated "greeting card" with music. Both can contain hyperlinks.

The homebuilders begin by selecting the geographic region of the state that they want to access. Then, the homebuilders enter the "headline" which will go into the subject field. Next, they enter the text message, and email address for return replies. The homebuilders then enter their usernames and passwords and click the "mail to realtors" button for immediate and private electronic transmission of selected information to targeted professionals.

A program, such as a CGI program, then bulk emails the message to all of the real estate professionals who have added themselves to the list for that region. If the homebuilders prefer to use animated cards, they can select which graphics they would like to send and then complete the same information as indicted above.

Example

An example of an embodiment of the present invention as set forth on the following pages will now be further illustrated. It should be apparent to those of skill in this art that the programs and supporting files utilized to carry out the objectives of this invention can be loaded onto any suitable device or computer hardware and can be formatted for compatibility with any server and search engine.

Section I

This section will contain the CGI programs and supporting files that are used at <http://www.brokerswelcome.com>.

The following nine CGI programs were written in PERL, they must be stored in the cgi-bin folder located on the web server and must have read and execute permissions:

1. **db-add.cgi**, this program is used to add the realtors to our flat file database:

```
#!/usr/bin/perl

require "process CGI.pl";
require "db-head.pl";

&parse_input(*fields);

&print_header;

$filePrefix=$fields{"FilePrefix"};
delete $fields{"FilePrefix"};

$filePrefix2=$filePrefix . "2";

$RowID=0;

if(exists $fields{"RowID"}) {
    $RowID=$fields{"RowID"};
    delete $fields{"RowID"};
}

if(exists $fields{"Area"}) {
    if ($fields{"Area"} eq "Statewide") {
        $fields{"City"}="Statewide";
    }
}

if(exists $fields{"RequiredFields"}) {
    @rfields=split(/&/,$temp);
    $match=0;
    foreach $fname(@rfields) {
        if(!(exists $fields{$fname})) {
            $match=1;
        }
    }
    if($match==1) {
        open FAILURE, "<". $filebase.$filePrefix."-fields.html";
        while(<FAILURE>) {
            print;
        }
        exit(0);
    }
}
```

```

if (&validate_user==0) {
open FAILURE, "<". $filebase.$filePrefix."-failure.html";
while (<FAILURE>) {
print;
}
exit(0);
}

```

```

$fields{"RowID"}=time;

```

```

foreach $key (keys %fields) {
$fields{$key} =~ s/\n/<BR>/g;
$encode=$fields{$key};
$encode=&encode_string($encode);
$g=$g."$key=$encode&";
}

```

```

chop $g;

```

```

if($RowID!=0) {
$searchString="RowID=$RowID";
open LISTINGS, "<". $filebase.$filePrefix."-listings.lis";
open LISTBUF, ">". $filebase.$filePrefix."-listings.buf";

```

```

while (<LISTINGS>) {
if(/$searchString/)
{$Match=0;} else {$Match=1;}

```

```

if($Match==1) {
print LISTBUF $_;
}
}

```

```

close LISTBUF;
close LISTINGS;

```

```

open LISTBUF, "<". $filebase.$filePrefix."-listings.buf";
open LISTINGS, ">". $filebase.$filePrefix."-listings.lis";

```

```

while(<LISTBUF>) {
print LISTINGS $_;
}

```

```

close LISTINGS;
close LISTBUF;

```

```

if (-e "$picturebase/$filePrefix/$RowID.gif") {
`mv $picturebase/$filePrefix/$RowID.gif $picturebase/$filePrefix/$fields{RowID}.gif`;
}
if (-e "$picturebase/$filePrefix/$RowID.jpg") {
`mv $picturebase/$filePrefix/$RowID.jpg $picturebase/$filePrefix/$fields{RowID}.jpg`;
}
if (-e "$picturebase/$filePrefix2/$RowID.gif") {
`mv $picturebase/$filePrefix2/$RowID.gif $picturebase/$filePrefix2/$fields{RowID}.gif`;
}

```



```

if (-e "$picturebase/$filePrefix2/$RowID.jpg") {
`mv $picturebase/$filePrefix2/$RowID.jpg $picturebase/$filePrefix2/$fields{RowID}.jpg`;
}
}

open LISTINGS, ">>".$filebase.$filePrefix."-listings.lis";
print LISTINGS $g."n";
close LISTINGS;

open SUCCESS, "<".$filebase.$filePrefix."-success.html";

while(<SUCCESS>) {
print;
}

sub AddPassword {
open PASS, ">>".$filebase.$PassFile;
print PASS "$fields{CustomUserName}:$fields{CustomPassword}:None:$fields{Contact_Name}\n";
close PASS;
}

```

2. **db-list.cgi**, this program is used to query the database, it could also be used to bulk email the realtors:

```

#!/usr/bin/perl

require "process CGI.pl";
require "db-head.pl";

$|=1;

sub parse_string {
chop;
$parseme=shift(@_);
@pairs=split(/&/,$parseme);
foreach $item(@pairs) {
($key,$content)=split(/=/,$item,2);
$content=&decode_string($content);
$content=~s/\n/<BR>/g;
$key=&decode_string($key);
$input{$key}=$content;
}
}

sub print_record {
foreach $line (@Template) {
$Temp=$line;
foreach $key (keys %input) {
$Temp=~s/\$\\$key\\$\\$/input{$key}/ge;
}
$Temp=~s/\$\\$(.*?)\\$\\$/g;
foreach $key (keys %input) {
$ckey=&encode_string(input{$key});
$Temp=~s/\#\\$key\\#\\$/ckey/ge;
}
$Temp=~s/\#\\$(.*?)\\#\\$/g;

```

```

print $Temp;
}
}

```

```

&print_header;

```

```

&parse_input(*fields);
$filePrefix=$fields{"FilePrefix"};
delete $fields{"FilePrefix"};

```

```

$filePrefix2=$filePrefix . "2";

```

```

$FileHeader=$filePrefix."-header.html";

```

```

if(exists $fields{Header}) {
$FileHeader=$fields{Header};
delete $fields{Header};
}

```

```

$FileFooter=$filePrefix."-footer.html";

```

```

if(exists $fields{Footer}) {
$FileFooter=$fields{Footer};
delete $fields{Footer};
}

```

```

$FileNoneFound="generic-none.html";

```

```

if(exists $fields{NoneFound}) {
$FileNoneFound=$fields{NoneFound};
delete $fields{NoneFound};
}

```

```

if(exists $fields{InFrame}) {
print "<HTML><HEAD><TITLE>\n";
print "$fields{InFrame}\n";
print "</TITLE></HEAD><BODY>\n";
delete $fields{InFrame};
$InFrame=1;
} else {
open HEADER, "<" . $filebase . $FileHeader;
while(<HEADER>) {
print;
}
close HEADER;
}

```

```

if(exists $fields{ToMailList}) {
$ToMailList=1;
delete $fields{ToMailList};
$FormFrom = $fields{FormFrom};
$FormSubject = $fields{FormSubject};
$FormText = $fields{FormText};
delete $fields{FormFrom};
delete $fields{FormSubject};
}

```

```
delete $fields{FormText};
}
```

```
$validated=0;
```

```
if(exists $fields{UserPassword}) {
    $validated=&validate_user;
    if (not($validated)) {
        print "<H1>Invalid Password</H1>\n";
        open FOOTER, "<".$filebase.$FileFooter;
        while(<FOOTER>) {
            print;
        }
    }
    if($fields{UserName} eq 'All') { delete $fields{UserName} };
    if($ToMailList == 1) { delete $fields{UserName} };
    delete $fields{UserPassword};
    delete $fields{UserURL};
    delete $fields{UserFullName};
}
```

```
if(exists $fields{HasPhotos}) {
    $HasPhotos=1;
    delete $fields{HasPhotos}
}
```

```
if(exists $fields{HasPhotos2}) {
    $HasPhotos2=1;
    delete $fields{HasPhotos2}
}
```

```
if(exists $fields{CanEdit}) {
    $CanEdit=1;
    delete $fields{CanEdit}
}
```

```
foreach $key (keys %fields) {
    if($fields{$key} eq 'All') {delete $fields{$key};}
    if($fields{$key} eq '') {delete $fields{$key};}
}
```

```
if(exists $fields{Price}) {
    $fields{Price} =~ s/\$/g;
    $fields{Price} =~ s/,/g;
}
```

```
if(exists $fields{PriceLow}) {
    $fields{PriceLow} =~ s/\$/g;
    $fields{PriceLow} =~ s/,/g;
    $PriceLow=$fields{PriceLow};
    delete $fields{PriceLow};
    $PriceHigh=999999999;
} else {
    $PriceLow="None";
}
```

```

if(exists $fields{PriceHigh}) {
$fields{PriceHigh}~~s/^$/g;
$fields{PriceHigh}~~s/./g;
$PriceHigh=$fields{PriceHigh};
delete $fields{PriceHigh};
}

```

```

if(exists $fields{StartPrice}) {
$StartPrice=$fields{StartPrice};
$StartPrice~~s/./g;
$StartPrice~~s/^$/g;
$EndPrice=999999999;
delete $fields{StartPrice};
} else {
$StartPrice="None";
}

```

```

if(exists $fields{EndPrice}) {
$EndPrice=$fields{EndPrice};
$EndPrice~~s/./g;
$EndPrice~~s/^$/g;
delete $fields{EndPrice};
}

```

```

if(exists $fields{HeaderText}) {
print $fields{HeaderText};
delete $fields{HeaderText};
}

```

```

if((exists $fields{CustomPassword}) and (exists $fields{Email})) {
$CustomPassword=$fields{CustomPassword};
$Email=$fields{Email};
}

```

```

$FileTemplate=$filePrefix."-template.html";

```

```

if(exists $fields{Template}) {
$FileTemplate=$fields{Template};
delete $fields{Template};
}

```

```

open TEMPLATE, "<".$filebase.$FileTemplate;
@Template=<TEMPLATE>;
close TEMPLATE;

```

```

open LISTINGS, "<".$filebase.$filePrefix."-listings.lis";

```

```

$record_count=0;

```

b

```

while (<LISTINGS>) {

```

```

$Match=0;

```

```

foreach $key (keys %fields) {
$searchString=&encode_string($fields{$key});
$searchKey=&encode_string($key);

```

```

$searchString="$searchKey=$searchString";
$searchString=~s/\$/\\$/g;
if($Match==0) {
  if(/$searchString/i)
    {$Match=0;} else {$Match=1;}
  if($key eq "Area") {
    if(/Statewide/) {$Match=0;}
  }
}
}
}

```

```

if ($Match==0) {
  %input=();
  &parse_string($_);
}

```

```

if ($Match==0) {
  if ($StartPrice ne "None") {
    $price=$input{Price};
    $price=~s/./g;
    $price=~s/\$/g;
    if ($price<$StartPrice) { $Match=1; }
    if ($price>$EndPrice) { $Match=1; }
  }
}

```

```

if ($PriceLow ne "None") {
  $LPrice=$input{PriceLow};
  $LPrice=~s/./g;
  $LPrice=~s/\$/g;
}

```

```

$HPrice=$input{PriceHigh};
$HPrice=~s/./g;
$HPrice=~s/\$/g;

```

```

if (($PriceHigh>=$LPrice) && ($PriceLow<=$HPrice)) {
  $Match=0;
} else { $Match=1; }
}

```

```

}

```

```

if ($Match==0) {
  if($CustomPassword) {
    if ( ($CustomPassword eq $input{CustomPassword}) and
      ($Email eq $input{Email}) ) {
      $validated=1;
    } else { $validated=0; $Match=1; }
  }
}

```

```

if($validated) {
  print '<br><a href="'. $cgiurlbase.'db-delete.cgi?FilePrefix='.$filePrefix.'&RowID='.$input{RowID}.'">Click to
    Delete</a>';
  if ($CanEdit) {
    print '<br><a href="'. $cgiurlbase.'db-edit.cgi?FilePrefix='.$filePrefix.'&RowID='.$input{RowID}.'">Click to
      Edit</a>';
  }
}

```

```

}
if($HasPhotos) {
print '<br><a href="'. $cgiurlbase.'db-download.cgi?FilePrefix='.$filePrefix.'&RowID='.$Sinut{RowID}.'">Click
to Upload Photo</a>';
}
if($HasPhotos2) {
print '<br><a href="'. $cgiurlbase.'db-download2.cgi?FilePrefix='.$filePrefix.'&RowID='.$Sinut{RowID}.'">Click
to Upload Second Photo</a>';
}
}
if($HasPhotos) {
if(-e "$picturebase/$filePrefix/$Sinut{RowID}.gif") {
Sinut{picture}=<a href="'. $pictureurl/$filePrefix/";
Sinut{picture}=$Sinut{RowID}.'.gif';
Sinut{picture}="><IMG width=73 height=51 BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picture}=$Sinut{RowID}.'.gif';
Sinut{picture}="></a>";
Sinut{picfilepath}=<IMG BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picfilepath}=$Sinut{RowID}.'.gif';
Sinut{picfilepath}=">";
Sinut{picfilepathleft}=<IMG BORDER=0 align=left SRC="'. $pictureurl/$filePrefix/";
Sinut{picfilepathleft}=$Sinut{RowID}.'.gif';
Sinut{picfilepathleft}=">";
Sinut{picsmallpath}=<IMG width=73 height=51 BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallpath}=$Sinut{RowID}.'.gif';
Sinut{picsmallpath}=">";
Sinut{picsmallleft}=<a href="'. $pictureurl/$filePrefix/";
Sinut{picsmallleft}=$Sinut{RowID}.'.gif';
Sinut{picsmallleft}="><IMG width=73 height=51 align=left BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallleft}=$Sinut{RowID}.'.gif';
Sinut{picsmallleft}="></a>";
Sinut{picsmallright}=<IMG width=73 height=51 align=right BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallright}=$Sinut{RowID}.'.gif';
Sinut{picsmallright}=">";
Sinut{picsmallfpm}=<IMG width=219 height=153 BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallfpm}=$Sinut{RowID}.'.gif';
Sinut{picsmallfpm}=">";
} elseif (-e "$picturebase/$filePrefix/$Sinut{RowID}.jpg") {
Sinut{picture}=<a href="'. $pictureurl/$filePrefix/";
Sinut{picture}=$Sinut{RowID}.'.jpg';
Sinut{picture}="><IMG width=73 height=51 BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picture}=$Sinut{RowID}.'.jpg';
Sinut{picture}="></a>";
Sinut{picfilepath}=<IMG BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picfilepath}=$Sinut{RowID}.'.jpg';
Sinut{picfilepath}=">";
Sinut{picfilepathleft}=<IMG BORDER=0 align=left SRC="'. $pictureurl/$filePrefix/";
Sinut{picfilepathleft}=$Sinut{RowID}.'.jpg';
Sinut{picfilepathleft}=">";
Sinut{picsmallpath}=<IMG width=73 height=51 BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallpath}=$Sinut{RowID}.'.jpg';
Sinut{picsmallpath}=">";
Sinut{picsmallleft}=<a href="'. $pictureurl/$filePrefix/";
Sinut{picsmallleft}=$Sinut{RowID}.'.jpg';
Sinut{picsmallleft}="><IMG width=73 height=51 align=left BORDER=0 SRC="'. $pictureurl/$filePrefix/";
Sinut{picsmallleft}=$Sinut{RowID}.'.jpg';

```

```

Sinput{picsmallleft}.="</a>";
Sinput{picsmallright}="<IMG width=73 height=51 align=right BORDER=0 SRC="."$pictureurl/$filePrefix/";
Sinput{picsmallright}=$Sinput{RowID}.'.jpg';
Sinput{picsmallright}.=">";
Sinput{picsmallfpm}="<IMG width=219 height=153 BORDER=0 SRC="."$pictureurl/$filePrefix/";
Sinput{picsmallfpm}=$Sinput{RowID}.'.jpg';
Sinput{picsmallfpm}.=">";
}
}
if($HasPhotos2) {
if(-e "$picturebase/$filePrefix2/$Sinput{RowID}.gif") {
Sinput{picture2}="<a href="."$pictureurl/$filePrefix2/";
Sinput{picture2}=$Sinput{RowID}.'.gif';
Sinput{picture2}="><IMG width=73 height=51 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picture2}=$Sinput{RowID}.'.gif';
Sinput{picture2}="></a>";
Sinput{picfilepath2}="<IMG BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picfilepath2}=$Sinput{RowID}.'.gif';
Sinput{picfilepath2}=">";
Sinput{picfilepathleft2}="<IMG BORDER=0 align=left SRC="."$pictureurl/$filePrefix2/";
Sinput{picfilepathleft2}=$Sinput{RowID}.'.gif';
Sinput{picfilepathleft2}=">";
Sinput{picsmallpath2}="<IMG width=73 height=51 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallpath2}=$Sinput{RowID}.'.gif';
Sinput{picsmallpath2}=">";
Sinput{picsmallleft2}="<a href="."$pictureurl/$filePrefix2/";
Sinput{picsmallleft2}=$Sinput{RowID}.'.gif';
Sinput{picsmallleft2}="><IMG width=73 height=51 align=left BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallleft2}=$Sinput{RowID}.'.gif';
Sinput{picsmallleft2}="></a>";
Sinput{picsmallright2}="<IMG width=73 height=51 align=right BORDER=0 SRC="."$pictureurl/$filePrefix/";
Sinput{picsmallright2}=$Sinput{RowID}.'.gif';
Sinput{picsmallright2}=">";
Sinput{picsmallfpm2}="<IMG width=219 height=153 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallfpm2}=$Sinput{RowID}.'.gif';
Sinput{picsmallfpm2}=">";
} elseif (-e "$picturebase/$filePrefix2/$Sinput{RowID}.jpg") {
Sinput{picture2}="<a href="."$pictureurl/$filePrefix2/";
Sinput{picture2}=$Sinput{RowID}.'.jpg';
Sinput{picture2}="><IMG width=73 height=51 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picture2}=$Sinput{RowID}.'.jpg';
Sinput{picture2}="></a>";
Sinput{picfilepath2}="<IMG BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picfilepath2}=$Sinput{RowID}.'.jpg';
Sinput{picfilepath2}=">";
Sinput{picfilepathleft2}="<IMG BORDER=0 align=left SRC="."$pictureurl/$filePrefix2/";
Sinput{picfilepathleft2}=$Sinput{RowID}.'.jpg';
Sinput{picfilepathleft2}=">";
Sinput{picsmallpath2}="<IMG width=73 height=51 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallpath2}=$Sinput{RowID}.'.jpg';
Sinput{picsmallpath2}=">";
Sinput{picsmallleft2}="<a href="."$pictureurl/$filePrefix2/";
Sinput{picsmallleft2}=$Sinput{RowID}.'.jpg';
Sinput{picsmallleft2}="><IMG width=73 height=51 align=left BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallleft2}=$Sinput{RowID}.'.jpg';
Sinput{picsmallleft2}="></a>";

```

```

Sinput{picsmallright2}=<IMG width=73 height=51 align=right BORDER=0 SRC="."$pictureurl/$filePrefix/";
Sinput{picsmallright2}=$Sinput{RowID}.'.jpg';
Sinput{picsmallright2}.=">";
Sinput{picsmallfpm2}=<IMG width=219 height=153 BORDER=0 SRC="."$pictureurl/$filePrefix2/";
Sinput{picsmallfpm2}=$Sinput{RowID}.'.jpg';
Sinput{picsmallfpm2}.=">";
}
}
}

```

```

if ($Match==0) {
Sinput{Price}=-s/$//g;
Sinput{Price}=$Sinput{Price};
&print_record;
if ($ToMailList == 1) {
$FormTo = Sinput{Email};
&SendMail;
}
$record_count=1;
}
}

```

```

if($record_count!=1) {
open EMPTY, "<". $filebase.$FileNoneFound;
while(<EMPTY>) {
print;
}
close EMPTY;
}
close LISTINGS;

```

```

if($InFrame) {
print "</BODY></HTML>\n";
} else {
open FOOTER, "<". $filebase.$FileFooter;

```

```

while(<FOOTER>) {
print;
}

```

```

close FOOTER;
}

```

```

sub SendMail {

```

```

open (SENDMAIL, "| /usr/sbin/sendmail -t -n -oi");

```

```

print SENDMAIL "From: $FormFrom\n";
print SENDMAIL "To: $FormTo\n";
print SENDMAIL "Reply-To: $FormFrom\n";
print SENDMAIL "Subject: $FormSubject\n";

```

```

print SENDMAIL "$FormText\n";

```

```

close SENDMAIL;
}

```


3. **db-edit.cgi**, this program is used to edit the realtor listings and the animated cards:

```
#!/usr/bin/perl
```

```
require "process CGI.pl";
require "db-head.pl";
```

```
sub parse_string {
    chop;
    $parseme=shift(@_);
    @pairs=split(/&/,$parseme);
    foreach $item(@pairs) {
        ($key,$content)=split(/=/,$item,2);
        $content=&decode_string($content);
        $content=~s/\n/<BR>\n/g;
        $key=&decode_string($key);
        $input{$key}=$content;
    }
}
```

```
sub print_record {
    foreach $line (@Template) {
        $Temp=$line;
        foreach $key (keys %input) {
            $Temp=~s/\$\$key\$\$/input{$key}/ge;
        }
        $Temp=~s/\$\$(.*?)\$/\$/g;
        print $Temp;
    }
}
```

```
&print_header;
&parse_input(*fields);
$filePrefix=$fields{"FilePrefix"};
delete $fields{"FilePrefix"};
```

```
$validated=0;
```

```
if(exists $fields{UserPassword}) {
    $validated=&validate_user;
    if (not($validated)) {
        print "<H1>Invalid Password</H1>\n";
        open FOOTER, "<". $filebase.$filePrefix."-footer.html";
        while(<FOOTER>) {
            print;
        }
    }
    if($fields{UserName} eq 'All') { delete $fields{UserName} };
    delete $fields{UserPassword};
    delete $fields{UserURL};
    delete $fields{UserFullName};
}
```

```
if(exists $fields{HasPhotos}) {
    $HasPhotos=1;
```

```

delete $fields{HasPhotos}
}

foreach $key (keys %fields) {
if($fields{$key} eq 'All') {delete $fields{$key};}
if($fields{$key} eq ' ') {delete $fields{$key};}
}

if(exists $fields{Price}) {
$fields{Price}~s/$//g;
$fields{Price}~s/./g;
}

if(exists $fields{StartPrice}) {
$StartPrice=$fields{StartPrice};
$StartPrice~s/./g;
$StartPrice~s/$//g;
$EndPrice=999999999;
delete $fields{StartPrice};
} else {
$StartPrice="None";
}

if(exists $fields{EndPrice}) {
$EndPrice=$fields{EndPrice};
$EndPrice~s/./g;
$EndPrice~s/$//g;
delete $fields{EndPrice};
}

if(exists $fields{HeaderText}) {
print $fields{HeaderText};
delete $fields{HeaderText};
}

if((exists $fields{CustomPassword}) and (exists $fields{Email})) {
$CustomPassword=$fields{CustomPassword};
$Email=$fields{Email};
}

open TEMPLATE, "<".$filebase.$filePrefix."-edit.html";
@Template=<TEMPLATE>;
close TEMPLATE;

open LISTINGS, "<".$filebase.$filePrefix."-listings.lis";

while (<LISTINGS>) {

$Match=0;

foreach $key (keys %fields) {
$searchString=&encode_string($fields{$key});
$searchKey=&encode_string($key);
$searchString="$searchKey=$searchString";
$searchString~s/$$\\$/g;
if($Match==0) {

```

```

if(/$searchString/i)
{$Match=0;} else {$Match=1;}
}
}

```

```

if ($Match==0) {
%input=();
&parse_string($_);
}

```

```

if ($Match==0) {
if ($StartPrice ne "None") {
$price=$input{Price};
$price=~s/./g;
$price=~s/\$/g;
if ($price<$StartPrice) { $Match=1; }
if ($price>$EndPrice) { $Match=1; }
}
}

```

```

if ($Match==0) {
if($CustomPassword) {
if ( ($CustomPassword eq $input{CustomPassword}) and
($Email eq $input{Email})) {
$validated=1;
} else { $validated=0; $Match=1; }
}
}

```

```

if($validated) {
print '<a href="'. $cgiurlbase.'db-delete.cgi?FilePrefix='.$filePrefix.'&RowID='.$input{RowID}.'">Click to Delete
this Listing</a><br>';
print '<a href="'. $cgiurlbase.'db-edit.cgi?FilePrefix='.$filePrefix.'&RowID='.$input{RowID}.'">Click to Edit this
Listing</a><br>';
if($HasPhotos) {
print '<br><a href="'. $cgiurlbase.'db-download.cgi?FilePrefix='.$filePrefix.'&RowID='.$input{RowID}.'">Click
to Upload Photo</a> ';
}
}
}

```

```

if($HasPhotos) {
if(-e "$picturebase/$filePrefix/$input{RowID}.gif") {
$input{picture}='<a href="'. $pictureurl/$filePrefix/";
$input{picture}=$input{RowID}.'.gif';
$input{picture}.='<IMG BORDER=0 SRC="'. $pictureurl/$filePrefix/pic-icon.gif'."></a>';
} elsif (-e "$picturebase/$filePrefix/$input{RowID}.jpg") {
$input{picture}='<a href="'. $pictureurl/$filePrefix/";
$input{picture}=$input{RowID}.'.jpg';
$input{picture}.='<IMG BORDER=0 SRC="'. $pictureurl/$filePrefix/pic-icon.gif'."></a>';
}
}
}

```

```

if ($Match==0) {
$input{Price}::~s/\$/g;
$input{Price}='$'. $input{Price};
&print_record;
}
}

```

```

close LISTINGS;

```

4. process cgi.pl, this program is used to handle all the variables that are being passed to the other CGI programs:

```
#!/usr/bin/perl
```

```
sub encode_string {
    local($encodes);
    $encodes=shift(@_);
    $encodes=~s/^%/%25/g;
    $encodes=~s/^+/%2b/g;
    $encodes=~s/^&/%26/g;
    $encodes=~s/ /%20/g;
    $encodes=~s/^=/%3d/g;
    $encodes=~s/^r//g;
    $encodes=~s/^n/%0d/g;
    return $encodes;
}

sub decode_string {
    local($decodes);
    $decodes=shift(@_);
    $decodes=~tr/+//;
    $decodes=~s/%(..)/pack("c",hex($1))/ge;
    return $decodes;
}

sub form_method {
    $method=$ENV{'REQUEST_METHOD'};
}

sub path_info {
    $path=$ENV{'PATH_INFO'};
    $path=~s/^//;
    $return=$path;
}

sub print_header {
    if (!defined(@_)) {
        print "Content-type: text/html\n\n";
    }
    else {
        print "Location: @_ \n\n";
    }
}

sub parse_input {
    local(*input)=shift(@_);

    local($temp,@pairs);
    if(&form_method eq 'POST') {
        read(STDIN,$temp,$ENV{'CONTENT_LENGTH'});
    }
    else {
        $temp=$ENV{'QUERY_STRING'};
    }
}
```

```

@pairs=split(/&/,$temp);
foreach $item(@pairs) {
($key,$content)=split(/=/,$item,2);
$content=&decode_string($content);
$key=&decode_string($key);
if (!defined($input{$key})) {
$input{$key}=$content;
$input[@input]=$key;
}
else {
$input{$key}.="|$content";
}
}
return 1;
}

return 1;

```

5. **db-head.pl**, this program is used to set the local paths to the cgi-files directory and to the pictures directory.

```

$filebase="/home/ahguides/cgi-files";
$filebase.="";

$cgiurlbase="/cgi-bin";
$cgiurlbase.="";

$picturebase="/home/ahguides/pictures";
$picturebase.="";

$pictureurl="/pictures";
$pictureurl.="";

$maxsize=54000;

sub validate_user {
open PASSWORDS, "<".$filebase.$filePrefix."-pass.lis";
while(<PASSWORDS>) {
($UserName,$Password,$Path,$UserFullName)=split /\:/, $_, 4;
if ($UserName eq $fields{"UserName"}) {
if ($Password eq $fields{"UserPassword"}) {
$fields{"UserURL"}=$Path;
$fields{"UserFullName"}=$UserFullName;
return 1;
}
}
}
return 0;
}

return 1;

```

6. **db-delete.cgi**, this program is used to remove entries from the database:

```

#!/usr/bin/perl

```

```
require "process_cgi.pl";
require "db-head.pl";
```

```
sub parse_string {
  $parseme=shift(@_);
  @pairs=split(/&/,$parseme);
  foreach $item(@pairs) {
    ($key,$content)=split(/=/,$item,2);
    $content=&decode_string($content);
    $content=~s/\n/<BR>\n/g;
    $key=&decode_string($key);
    $input{$key}=$content;
  }
}
```

```
sub print_record {
  foreach $line (@Template) {
    $Temp=$line;
    foreach $key (keys %input) {
      $Temp=~s/\$\$key\$\$/input{$key}/ge;
    }
    $Temp=~s/\$\$(.*?)\$/\$/g;
    print $Temp;
  }
}
```

```
&print_header;
&parse_input(*fields);
$filePrefix=$fields{"FilePrefix"};
delete $fields{"FilePrefix"};
```

```
$filePrefix2=$filePrefix . "2";
```

```
open HEADER, "<". $filebase.$filePrefix."-header.html";
while(<HEADER>) {
  print;
}
close HEADER;
```

```
$validated=0;
print "Removing Listing....<br>\n";
open TEMPLATE, "<". $filebase.$filePrefix."-template.html";
@Template=<TEMPLATE>;
close TEMPLATE;
open LISTINGS, "<". $filebase.$filePrefix."-listings.lis";
open LISTBUF, ">". $filebase.$filePrefix."-listings.buf";
$Matches=0;
while (<LISTINGS>) {
  $Match=0;
  foreach $key (keys %fields) {
    $searchString=&encode_string($fields{$key});
    $searchKey=&encode_string($key);
    $searchString="$searchKey=$searchString";
    if($Match==0) {
      if($searchString)
        {$Match=0;} else {$Match=1;}
    }
  }
}
```

```

}
}

if ($Match==0) {
%input=();
&parse_string($_);
&print_record;
if (-e "$picturebase/$filePrefix/$input{RowID}.gif") {
`rm $picturebase/$filePrefix/$input{RowID}.gif`;
print "<br>Picture Removed<br>\n";
}
if (-e "$picturebase/$filePrefix/$input{RowID}.jpg") {
`rm $picturebase/$filePrefix/$input{RowID}.jpg`;
print "<br>Picture Removed<br>\n";
}
if (-e "$picturebase/$filePrefix2/$input{RowID}.gif") {
`rm $picturebase/$filePrefix2/$input{RowID}.gif`;
print "<br>Picture Removed<br>\n";
}
if (-e "$picturebase/$filePrefix2/$input{RowID}.jpg") {
`rm $picturebase/$filePrefix2/$input{RowID}.jpg`;
print "<br>Picture Removed<br>\n";
}
$Matches++;
} else {
print LISTBUF $_;
}
}

close LISTINGS;
close LISTBUF;

open LISTBUF, "<". $filebase.$filePrefix."-listings.buf";
open LISTINGS, ">". $filebase.$filePrefix."-listings.lis";

while(<LISTBUF>) {
print LISTINGS $_;
}

close LISTINGS;
close LISTBUF;

print "$Matches Listings Deleted<br>\n";

open FOOTER, "<". $filebase.$filePrefix."-footer.html";

while(<FOOTER>) {
print;
}

close FOOTER;

```

7. **db-extend.cgi**, this program is used to extend the entries in the database another 6 months:

```
#!/usr/bin/perl
```

```
require "process CGI.pl";
```

```

require "db-head.pl";

sub parse_string {
    $parseme=shift(@_);
    @pairs=split(/&/,$parseme);
    foreach $item(@pairs) {
        ($key,$content)=split(/=/,$item,2);
        $content=&decode_string($content);
        $content=~s/\n/<BR>\n/g;
        $key=&decode_string($key);
        $input{$key}=$content;
    }
}

sub print_record {
    foreach $line (@Template) {
        $Temp=$line;
        foreach $key (keys %input) {
            $Temp=~s/\$\$key\$\$/input{$key}/ge;
        }
        $Temp=~s/\$\$(.*?)\$/\$/g;
        print $Temp;
    }
}

&print_header;
&parse_input(*fields);
$filePrefix=$fields{"FilePrefix"};
delete $fields{"FilePrefix"};

open HEADER, "<".$filebase.$filePrefix."-header.html";
while(<HEADER>) {
    print;
}
close HEADER;

$validated=0;

print "Extending Listing....<br>\n";

open TEMPLATE, "<".$filebase.$filePrefix."-template.html";
@Template=<TEMPLATE>;
close TEMPLATE;

open LISTINGS, "<".$filebase.$filePrefix."-listings.lis";
open LISTBUF, ">".$filebase.$filePrefix."-listings.buf";

while (<LISTINGS>) {

    $Match=0;

    foreach $key (keys %fields) {
        $searchString=&encode_string($fields{$key});
        $searchKey=&encode_string($key);
        $searchString="$searchKey=$searchString";
        if($Match==0) {

```



```

if(/$searchString/)
{$Match=0;} else {$Match=1;}
}
}

if ($Match==0) {
%input=();
&parse_string($_);
&print_record;
SoldRow=$input{"RowID"};
$input{"RowID"}=time;
if (-e "$picturebase/$filePrefix/$SoldRow.gif") {
`mv $picturebase/$filePrefix/$SoldRow.gif $picturebase/$filePrefix/$input{RowID}.gif`;
}
if (-e "$picturebase/$filePrefix/$SoldRow.jpg") {
`mv $picturebase/$filePrefix/$SoldRow.jpg $picturebase/$filePrefix/$input{RowID}.jpg`;
}

foreach $key (keys %input) {
$encode=$input{$key};
$encode=&encode_string($encode);
$g=$g."$key=$encode&";
}
chop $g;

print LISTBUF $g;
} else {
print LISTBUF $_;
}
}
close LISTINGS;
close LISTBUF;

open LISTBUF, "<". $filebase.$filePrefix."-listings.buf";
open LISTINGS, ">". $filebase.$filePrefix."-listings.lis";

while(<LISTBUF>) {
print LISTINGS $_;
}

close LISTINGS;
close LISTBUF;

print "Done!<br>\n";

open FOOTER, "<". $filebase.$filePrefix."-footer.html";

while(<FOOTER>) {
print;
}

close FOOTER;

```

8. **db-form.cgi**, is used to process variables that are passed to it from other pages for example got to <http://gator.twister.net/cgi-bin/db->

```
list.cgi?FilePrefix=com&BrokerProgram=on&Template=reg-com.tmpl&Header=reg-
header.html&Footer=reg-
footer.html&City=All&County=All&StartPrice=All&EndPrice=All&HomeType=All:
#!/usr/bin/perl
```

```
require "process_cgi.pl";
require "db-head.pl";
```

```
$|=1;
```

```
sub print_record {
  foreach $line (@Template) {
    $Temp=$line;
    foreach $key (keys %fields) {
      $Temp=~s/\$\$key\$\$/fields{$key}/ge;
    }
    $Temp=~s/\$\$(.*?)\$\$/g;
    foreach $key (keys %fields) {
      $ckey=&encode_string(fields{$key});
      $Temp=~s/\#\$key\#\#/$ckey/ge;
    }
    $Temp=~s/\#\$(.*?)\#\#/g;
    print $Temp;
  }
}
```

```
&print_header;
&parse_input(*fields);
```

```
if(exists $fields{Template}) {
  $FileTemplate=$fields{Template};
  delete $fields{Template};
} else {
  print "<HTML><BODY>Syntax Error in call to db-form.cgi</BODY></HTML>";
  exit;
}
```

```
open TEMPLATE, "<".$filebase.$FileTemplate;
@Template=<TEMPLATE>;
close TEMPLATE;
```

```
&print_record;
```

9. LYNOS bw_email.pl, is used to process forms and send the results via email to specified recipients:

```
#!/usr/bin/perl
```

```
use CGI;
$form = new CGI;
print $form->header;
```

```
##--//Form Field Values//-----##
```

```
$BuilderEmail = $form->param('BuilderEmail');
```

```

$BuilderEmail =~ tr/A-Za-z0-9\.\@\_\-//cds;

$ComName      = $form->param('CommunityName');

$ComPhone     = $form->param('CommunityPhone');
$ComURL       = $form->param('CommunityURL');

$ComPhone     = $form->param('CommunityPhone');
$ComURL       = $form->param('CommunityURL');

$B_Name       = $form->param('BrokerName');
$B_Name       =~ tr/A-Za-z0-9\.\@\_\-//cds;

$B_CoName     = $form->param('BrokerCompany');
$B_CoName     =~ tr/A-Za-z0-9\.\@\_\-//cds;

$B_Address    = $form->param('BrokerStreetAddress');
$B_Address    =~ tr/A-Za-z0-9\.\@\_\-//cds;

$B_City       = $form->param('BrokerCity');
$B_City       =~ tr/A-Za-z//cds;

$B_State      = $form->param('BrokerState');
$B_State      =~ tr/A-Za-z//cds;

$B_Zip        = $form->param('BrokerZip');
$B_Zip        =~ tr/0-9//cds;

$B_Phone      = $form->param('BrokerPhone');
$B_Phone      =~ tr/0-9//cds;
$F_PHONE1     = substr($B_Phone,0,3);
$F_PHONE2A    = substr($B_Phone,3,3);
$F_PHONE2B    = substr($B_Phone,6,4);
$B_Phone      = "(" . $F_PHONE1 . ")" . $F_PHONE2A . "-" . $F_PHONE2B;

$B_Fax        = $form->param('BrokerFax');
$B_Fax        =~ tr/0-9//cds;
$F_PHONE1     = substr($B_Fax,0,3);
$F_PHONE2A    = substr($B_Fax,3,3);
$F_PHONE2B    = substr($B_Fax,6,4);
$B_Fax        = "(" . $F_PHONE1 . ")" . $F_PHONE2A . "-" . $F_PHONE2B;

$B_Email      = $form->param('Email');
$B_Email      =~ tr/A-Za-z0-9\.\@\_\-//cds;

$C_Name       = $form->param('Name');
$C_Name       =~ tr/A-Za-z0-9\.\@\_\-//cds;

$C_Address    = $form->param('StreetAddress');
$C_Address    =~ tr/A-Za-z0-9\.\@\_\-//cds;

$C_City       = $form->param('City');
$C_City       =~ tr/A-Za-z//cds;

$C_State      = $form->param('State');
$C_State      =~ tr/A-Za-z//cds;

```

```
SC_Zip    = $form->param('Zip');
SC_Zip    =~ tr/0-9//cds;
```

```
SC_Phone  = $form->param('Phone');
SC_Phone  =~ tr/0-9//cds;
SF_PHONE1  = substr(SC_Phone,0,3);
SF_PHONE2A = substr(SC_Phone,3,3);
SF_PHONE2B = substr(SC_Phone,6,4);
SC_Phone   = "(" . SF_PHONE1 . ")" . SF_PHONE2A . "-" . SF_PHONE2B;
```

```
$FormTo    = $BuilderEmail;
$FormCc    = $B_Email;
$FormFrom  = "blynn\@gate.net";
$FormSubject = "Online Client Registration";
```

```
##--//Program Values//----- ##
$BaseDir = "/WWW/Web_Public_HTML/Web_Client_Docs/www.floridaguide.com/cgi-files-gator";
```

```
$FormHeader = "brokers-welcome_header.html";
$FormFooter = "brokers-welcome_footer.html";
```

```
## -- /////////////////////////////////////////////////// ----- ##
## -- /// START YOUR ENGINES /// ----- ##
## - /////////////////////////////////////////////////// ----- ##
&HTMLshooter($FormHeader);
&CheckEmail;
print "<CENTER>Thank You, your request has been processed.</CENTER>\n";
&HTMLshooter($FormFooter);
exit(0);
```

```
## -- /////////////////////////////////////////////////// ----- ##
## -- /// START YOUR ENGINES /// ----- ##
## - /////////////////////////////////////////////////// ----- ##
```

```
## -- Send Email Message ----- ##
## -- ----- ##
```

```
sub SendMail {
open (SENDMAIL, "|/usr/lib/sendmail -t -n -oi");
```

```
print SENDMAIL "From: $FormFrom\n";
print SENDMAIL "To: $FormTo\n";
print SENDMAIL "Cc: $FormCc\n";
print SENDMAIL "Reply-To: $FormFrom\n";
print SENDMAIL "Subject: $FormSubject\n";
```

```
print SENDMAIL "This is an online registration of a broker's client through Florida\n";
print SENDMAIL "Internet Real Estate Guide's Broker Participation Program. You have\n";
print SENDMAIL "agreed to honor registrations through this program for 30 days. If there\n";
print SENDMAIL "is any reason why this registration may not be honored, you must notify\n";
print SENDMAIL "the broker within 24 hours of receiving this registration or you will be\n";
print SENDMAIL "responsible for the brokerage commission at closing.\n";
print SENDMAIL "\n";
```

```

print SENDMAIL "Community Name: $ComName\n";
print SENDMAIL "Community Phone: $ComPhone\n";
print SENDMAIL "Communtiy Web Address: $ComURL\n";
print SENDMAIL "\n";
print SENDMAIL "-----\n";
print SENDMAIL "\n";
print SENDMAIL "Brokerage Name: $B_CoName\n";
print SENDMAIL "Broker Name: $B_Name\n";
print SENDMAIL "Broker Address: $B_Address\n";
print SENDMAIL "Broker City: $B_City\n";
print SENDMAIL "Broker State: $B_State\n";
print SENDMAIL "Broker Zip: $B_Zip\n";
print SENDMAIL "Broker Phone: $B_Phone\n";
print SENDMAIL "Broker E-mail: $B_Email\n";
print SENDMAIL "\n";
print SENDMAIL "-----\n";
print SENDMAIL "\n";
print SENDMAIL "Client Name: $C_Name\n";
print SENDMAIL "Client Address: $C_Address\n";
print SENDMAIL "Client City: $C_City\n";
print SENDMAIL "Client State: $C_State\n";
print SENDMAIL "Client Zip: $C_Zip\n";
print SENDMAIL "Client Phone: $C_Phone\n";
print SENDMAIL "\n";
close(SENDMAIL);
}

## -- Check Email Field ----- ##
## ----- ##
sub CheckEmail {

    $var2 = "Brokers Welcome";

    if ($B_Email =~ /([w-.]+@[w-.]+)/) {
        &SendMail;
    } else {
        print "Sorry, " . $var2 . " user. \n";
        print "Please enter a valid Email address.\n";
        print "<P>\n";
        print "This will allow us to send you the appropriate<BR>\n";
        print "information you have requested.\n";
        print "<P>\n";
        print "To continue press the <B>BACK</B> button on your browser.\n";
        &HTMLshooter($FormFooter);
        exit(0);
    }
}

## -- Send HTML shooter ----- ##
## ----- ##
sub HTMLshooter {
    open(SHOOTER,"<$BaseDir/$_[0]")
    || die "$_[0]: $!\n";
    while (<SHOOTER>) { # Print the body

```

```

print;
}
close SHOOTER;
##-- EXAMPLE CODE -- &HTMLshooter($html-file-name);
}

```

The following 25 files are required by the CGI programs. The .buf, .html, .lis, and .tmpl files, are the templates, headers, footers, and database files and must be stored in the cgi-files folder on the web server. The .buf and .lis files must have read and write permissions (example chmod 666), the other files only require read permissions (example chmod 655):

1. **brk-failure.html**, this is called upon if an error occurs while a realtor was attempting to add himself to the database:

```

<HTML>
<HEAD>
<TITLE>FloridaGuide Broker Participation Program</TITLE>
</HEAD>
<BODY background="images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
  BORDER=0 ALT="Rule">
<center></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>
<tr><td colspan="2" align="CENTER" valign="TOP"></td><td colspan="2" align="CENTER" valign="TOP">
</td></tr>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></td><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/register.html"></a></td><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/logolink.html"></a></td><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>
<p>
<H2 align="center"><font face="Arial">Unable To Add Listing.</font></H2>

```



```

E-mail to: <a
href="mailto:info@americanhomeguides.com">info@americanhomeguides.com</a></font><strong></strong>
<BR>
<br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>
<br>
</td>
</tr>
</table></center>
</BODY>
</HTML>

```

3. **brk-header.html**, this file is used as the header of all of the files that are generated by the CGI programs:

```

<HTML>
<HEAD>
<TITLE>FloridaGuide Broker Participation Program</TITLE>
</HEAD>
<BODY background="images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
BORDER=0 ALT="Rule">
<center></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>
<tr><td colspan="2" align="CENTER" valign="TOP"></TD><td colspan="2" align="CENTER" valign="TOP">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><a
href="http://www.brokerswelcome.com/register.html"></a></TD><td align="CENTER" valign="TOP"><a
href="http://www.brokerswelcome.com/logolink.html"></a></TD><td align="CENTER" valign="TOP"><a
href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>
<p>

```


<center>

4. **brk-listings.lis**, this is the flat file database below is example of one of the rows of data:

UserName=All&State=MA&Name=Deborah%20Baisley&UserURL=www.floridaguide.com&County=Norfolk&
UserPassword=fguide2100&UserFullName=Florida%20Internet%20Real%20Estate%20Guide
&Zip=020
38&Region=&RowID=918780214&Address=365%20West%20Central%20St.&Email=baisley@ma.ultranet.co
m&CompanyName=Dewolf&City=Franklin&Phone=508-520-0500

5. **brk-listings.buf**, this is a buffer for the database, it contains the same data as the database:

UserName=All&State=MA&Name=Deborah%20Baisley&UserURL=www.floridaguide.com&County=Norfolk&
UserPassword=fguide2100&UserFullName=Florida%20Internet%20Real%20Estate%20Guide
&Zip=020
38&Region=&RowID=918780214&Address=365%20West%20Central%20St.&Email=baisley@ma.ultranet.co
m&CompanyName=Dewolf&City=Franklin&Phone=508-520-0500

6. **brk-pass.lis**, this contains the usernames and passwords of all of the builders that are participating in this program, below is an example of one of the rows of data:

username:password:url:name

Robinette:rrh:www.floridaguide.com/robinettehomes:Robinette Homes

7. **brk-reg.html**, this is the registration file that is used when a realtor purchases an enhanced listing in Florida:

```
<HTML>
<HEAD>
<TITLE>Realtors - Real Estate Brokers Sign Up Program</TITLE>
<meta name="description" content="Get added to the e-mail list of area home builders who participate with
realtors. Simply fill out the information below. It's absolutely free.">
<meta name="keywords" content=", realtors, real estate, listings, realty, agents, realestate, brokers, broker listings
for real estate">
<script>
function FormSubmit()
{
if ( document.BrokerForm.Name.value == "" )
{
window.alert("Your Name must be entered before continuing.");
document.BrokerForm.Name.focus();
return;
}
if ( document.BrokerForm.CompanyName.value == "" )
{
window.alert("Your Company Name must be entered before continuing.");
document.BrokerForm.CompanyName.focus();
return;
}
if ( document.BrokerForm.County.value == "" )
{
window.alert("Your county must be entered before continuing.");
document.BrokerForm.County.focus();
return;
}
if ( document.BrokerForm.Address.value == "" )
{
window.alert("Your address must be entered before continuing.");
document.BrokerForm.Address.focus();
return;
}
}
```

```

if ( document.BrokerForm.City.value == "" )
{
window.alert("Your city must be entered before continuing.");
document.BrokerForm.City.focus();
return;
}
if ( document.BrokerForm.State.value == "" )
{
window.alert("Your state must be entered before continuing.");
document.BrokerForm.State.focus();
return;
}
if ( document.BrokerForm.Zip.value == "" )
{
window.alert("Your zip code must be entered before continuing.");
document.BrokerForm.Zip.focus();
return;
}
if ( document.BrokerForm.Phone.value == "" )
{
window.alert("Your telephone number must be entered before continuing.");
document.BrokerForm.Phone.focus();
return;
}
if ( document.BrokerForm.Email1.value == "" )
{
window.alert("Your E-Mail address must be entered before continuing.");
document.BrokerForm.Email1.focus();
return;
}
var emailadr=document.BrokerForm.Email1.value;
if ( emailadr.indexOf("@") < 1 )
{
window.alert("The e-mail address you have entered is invalid.");
document.BrokerForm.Email1.focus();
return;
}
document.BrokerForm.submit();
return;
}
</script>
</HEAD>
<BODY background="http://www.brokerswelcome.com/images/bckgd.gif" bgcolor="White" link="Red"
vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
BORDER=0 ALT="Rule">
<center><a href="http://www.brokerswelcome.com/index.html"></a></center>

```

<!-- STARBURST - REGISTER TO RECEIVE BUILDER INCENTIVES AND PROMOS -->

<FORM name="BrokerForm" action="http://gator.twister.net/cgi-bin/db-add.cgi" method="POST">
 <INPUT TYPE=HIDDEN NAME="FilePrefix" VALUE="brk">
 <INPUT TYPE=HIDDEN NAME="UserName" Value="All">
 <INPUT TYPE=HIDDEN NAME="UserPassword" Value="fguide2100">
 <INPUT TYPE="hidden" NAME="snd_total" VALUE="\$\$snd_total\$\$">
 <INPUT TYPE="hidden" NAME="Email" VALUE="\$\$snd_mail\$\$">
 <INPUT TYPE="hidden" NAME="snd_name" VALUE="\$\$snd_name\$\$">
 <INPUT TYPE="hidden" NAME="snd_addr1" VALUE="\$\$snd_addr1\$\$">
 <INPUT TYPE="hidden" NAME="snd_city" VALUE="\$\$snd_city\$\$">
 <INPUT TYPE="hidden" NAME="snd_state" VALUE="\$\$snd_state\$\$">
 <INPUT TYPE="hidden" NAME="snd_zip" VALUE="\$\$snd_zip\$\$">
 <INPUT TYPE="hidden" NAME="snd_country" VALUE="\$\$snd_country\$\$">
 <INPUT TYPE="hidden" NAME="snd_phone" VALUE="\$\$snd_phone\$\$">

<table border="0" width="420">
 <tr><td colspan="2" bgcolor="#FFCC66"><div
 align="center">ENHANCED LISTING

 ONLY \$29 FOR 6 MONTHS</div></td></tr>
 <tr>
 <td width="30%">
 Name:</td>
 <td width="70%">
 <input type="text" name="Name" size="61"></td>
 </tr>
 <tr>
 <td width="30%">
 Company Name:</td>
 <td width="70%">
 <input type="text" name="CompanyName" size="61"></td>
 </tr>
 <tr>
 <td width="30%">
 Region:</td>
 <td width="70%">Click Arrow to Select Region or Click Here for Regional Map.
 <small><select name="Region" size="1">
 <option value="">Please Select Your Region</option>
 <option value="Northeast: Jacksonville, St. Augustine, Gainesville">Northeast: Jacksonville, St. Augustine,
 Gainesville </option>
 <option value="Northwest: Tallahassee, Pensacola, Panama City">Northwest: Tallahassee, Pensacola, Panama
 City </option>
 <option value="East Central: Melbourne, Orlando, Daytona">East Central: Melbourne, Orlando, Daytona
 </option>
 <option value="South Central: Stuart, Port St. Lucie, Ft. Pierce, Vero Beach, Sebring, Lakeland">South Central:
 Stuart, Port St. Lucie, Ft. Pierce, Vero Beach, Sebring, Lakeland </option>
 <option value="West Central: Tampa, St. Petersburg, Ocala, Brooksville">West Central: Tampa, St. Petersburg,
 Ocala, Brooksville </option>
 <option value="Southwest: Naples, Ft. Myers, Sarasota, Punta Gorda">Southwest: Naples, Ft. Myers, Sarasota,
 Punta Gorda</option>
 <option value="Palm Beach: Boca Raton, Delray Beach, Boynton Beach, Palm Beach, Jupiter">Palm Beach:
 Boca Raton, Delray Beach, Boynton Beach, Palm Beach, Jupiter </option>


```

</FORM>
<CENTER><FONT FACE="Arial" SIZE="1">This form is intended for you to sign up to be on the builders'
    mailing list and become part of <A href="http://www.realtorlist.net">RealtorList.net</A>, a directory of
    realtors, real estate brokers and agents. It does not constitute registration of a client with any builder and is not
    an agreement by a builder to pay a commission. If you wish to register a client with a particular community, <A
    href="http://www.brokerswelcome.com/register.html">Click Here</A>.</FONT></CENTER>
<BR>
<IMG src="http://www.brokerswelcome.com/images/rule_bott.gif" WIDTH=585 HEIGHT=20 VSPACE=5
    BORDER=0 ALT="Rule">
</td>
</tr>
<!-- DISCLAIMER INFORMATION -->
<tr>
<TD></TD><td align="center"><DIV ALIGN="center"><TABLE><tr>
<td><A href="http://www.americanhomeguides.com/"></A></td>
<td><font face="arial" size="1">
Copyright &copy; 1999 American Home Guides, Inc.<br>
Phone: (954) 964-3717 Fax: (954) 966-7109<br>
E-mail to: <a
    href="mailto:info@americanhomeguides.com">info@americanhomeguides.com</a><br></font></td>
</tr></table>
<BR>
<br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>
<br>
</td>
</tr>
</table></center>
</BODY>
</HTML>

```

8. brk-success.html. This is the file that is called when a broker adds his listing successfully:

```

<HTML>
<HEAD>
<TITLE>FloridaGuide Broker Participation Program</TITLE>
</HEAD>
<BODY background="images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
    BORDER=0 ALT="Rule">
<center></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>

```

```
|  |  |  |  |
| --- | --- | --- | --- |
| Realtors | | Realtors | |
| Register to Receive Builder Incentives and Promos Register Your Clients with Builders Get Your Brokers Welcome Logo Link Mail Your Broker Program | | | |


## You are now registered with our program.



BOTTOM LINKS



Home Page
Register to Receive Builder Incentives and Promos
Register Your Clients with Builders



Get Your "Brokers Welcome" Logo Link
Mail Your Broker Program



![Rule](http://www.brokerswelcome.com/images/rule_bott.gif)



DISCLAIMER INFORMATION



Copyright &copy; 1999 Barry Lynn Marketing and Advertising, Inc.  

    Phone: (954) 964-2100 &nbsp; &nbsp; Fax: (954) 966-7109  

    E-mail to: info@americanhomeguides.com



Equal Housing Opportunity  

    Void in states where prohibited by law.


```

9. **brk-template.html**, this is the template that is used to display the realtor listings:


```

<HTML>
<HEAD>
<TITLE>FloridaGuide Broker Participation Program</TITLE>
</HEAD>
<BODY background="images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
  BORDER=0 ALT="Rule">
<center></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>
<tr><td colspan="2" align="CENTER" valign="TOP"></TD><td colspan="2" align="CENTER" valign="TOP">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/register.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/logolink.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>
<p>
<BR CLEAR=ALL>

```

12. reg-next.html, this the page where the broker enters in the information to register his client with the builder:

```

<HTML>
<HEAD>
<TITLE>FloridaGuide Broker Participation Program</TITLE>
<script>
function FormSubmit()
{
if ( document.Form.BrokerName.value == "" )
{
window.alert("Your Name must be entered before continuing.");
document.Form.BrokerName.focus();
return;
}
}

```



```

if ( document.Form.BrokerCompany.value == "" )
{
window.alert("Your Company Name must be entered before continuing.");
document.Form.BrokerCompany.focus();
return;
}
if ( document.Form.BrokerStreetAddress.value == "" )
{
window.alert("Your address must be entered before continuing.");
document.Form.BrokerStreetAddress.focus();
return;
}
if ( document.Form.BrokerCity.value == "" )
{
window.alert("Your city must be entered before continuing.");
document.Form.BrokerCity.focus();
return;
}
if ( document.Form.BrokerState.value == "" )
{
window.alert("Your state must be entered before continuing.");
document.Form.BrokerState.focus();
return;
}
if ( document.Form.BrokerZip.value == "" )
{
window.alert("Your zip code must be entered before continuing.");
document.Form.BrokerZip.focus();
return;
}
if ( document.Form.BrokerPhone.value == "" )
{
window.alert("Your Telephone Number must be entered before continuing.");
document.Form.BrokerPhone.focus();
return;
}
if ( document.Form.BrokerFax.value == "" )
{
window.alert("Your Fax Number must be entered before continuing.");
document.Form.BrokerFax.focus();
return;
}
if ( document.Form.Email.value == "" )
{
window.alert("Your E-Mail address must be entered before continuing.");
document.Form.Email.focus();
return;
}
var emailadr=document.Form.Email.value;
if ( emailadr.indexOf("@") < 1 )
{
window.alert("The e-mail address you have entered is invalid.");
document.Form.Email.focus();
return;
}
if ( document.Form.Name.value == "" )

```

```

{
window.alert("Your Client's Name must be entered before continuing.");
document.Form.Name.focus();
return;
}
if ( document.Form.StreetAddress.value == "" )
{
window.alert("Your Client's Address must be entered before continuing.");
document.Form.StreetAddress.focus();
return;
}
if ( document.Form.City.value == "" )
{
window.alert("Your Client's City must be entered before continuing.");
document.Form.City.focus();
return;
}
if ( document.Form.State.value == "" )
{
window.alert("Your Client's State must be entered before continuing.");
document.Form.State.focus();
return;
}
if ( document.Form.Zip.value == "" )
{
window.alert("Your Client's Zip code must be entered before continuing.");
document.Form.Zip.focus();
return;
}
if ( document.Form.Phone.value == "" )
{
window.alert("Your Client's Telephone Number must be entered before continuing.");
document.Form.Phone.focus();
return;
}
document.Form.submit();
return;
}
</script>
</HEAD>
<BODY background="images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG src="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
BORDER=0 ALT="Rule">
<center></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>

```

```

<tr><td colspan="2" align="CENTER" valign="TOP"></TD><td colspan="2" align="CENTER" valign="TOP">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/register.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/logolink.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>

```

```

<p>
<BR CLEAR=ALL>
<CENTER><FONT FACE="Arial" SIZE="+1" COLOR="#800000"><b>REGISTER YOUR CLIENTS WITH
  OUR<br>PARTICIPATING BUILDERS<br>
  FOR 30 DAYS OF PROTECTION</b></FONT><br>
<FONT FACE="Arial, Helvetica" SIZE="2"><B>Protect Yourself From the Risk of Your Clients<BR>
  Visiting a New Home Community Without You.</B></FONT></CENTER>
<P>
<FONT FACE="Arial, Helvetica" SIZE="2">Our participating builders will honor your online registration for 30
  days (assuming no prior registration exists). If your client arrives without you within 30 days, and executes a
  purchase and sale agreement, you will receive the comission that the builder normally pays per their Broker
  Participation Agreement.</FONT></P>
<P>
<FONT FACE="Arial, Helvetica" SIZE="2">A week before your registration expires, you will receive an e-mail
  reminder to renew your registration by visiting the community.</FONT></P>
<P>
<FONT FACE="Arial, Helvetica" SIZE="1"><I><B>Disclaimer:</B> If another broker brings in the person you
  registered, the builder will honor them as the procuring agent instead of you.</I></FONT></P>

```

```

<!-- Changed cgi url from /cgi-bin/db-add.cgi to /cgi-bin/LYNQS_bw_email.pl ----->
<FORM name="Form" action="http://gator.twister.net/cgi-bin/LYNQS_bw_email.pl" method="POST">

```

```

<INPUT TYPE=HIDDEN NAME="FilePrefix" VALUE="reg">
<INPUT TYPE=HIDDEN NAME="UserName" Value="All">
<INPUT TYPE=HIDDEN NAME="UserPassword" Value="fguide2100">
<INPUT TYPE=HIDDEN NAME="CommunityName" Value="$$CommunityName$$">
<!-- Added Hidden Field for BuilderEmail ----->
<INPUT TYPE=HIDDEN NAME="BuilderEmail" Value="$$BuilderEmail$$">
<INPUT TYPE=HIDDEN NAME="CommunityPhone" Value="$$CommunityPhone$$">
<INPUT TYPE=HIDDEN NAME="CommunityURL" Value="$$CommunityURL$$">
<!-- Community Information -->
<table border="0" width="420">
<TR><TD COLSPAN="2" bgcolor="#FFCC66">
<FONT FACE="Arial" SIZE="2"><B>COMMUNITY INFORMATION</B></FONT>
</TD></TR>
<tr>
<td>

```

```

<font face="Arial" size="2">Name:<br>
Phone#:<br>
Web Address:</font>
</td>
<td>
$$CommunityName$$<br>
$$CommunityPhone$$<br>
<a href="$$CommunityURL$$">$$CommunityURL$$</a>
</td>
</tr>
</table>
<!-- BROKER INFO -->
<table border="0" width="420">
<TR><TD COLSPAN="2" bgcolor="#FFCC66">
<FONT FACE="Arial" SIZE="2"><B>BROKER INFORMATION</B></FONT>
</TD></TR>
<tr>
<td>
<font face="Arial" size="2">Name:</font></td>
<td>
<input type="text" name="BrokerName" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Brokerage Company:</font></td>
<td>
<input type="text" name="BrokerCompany" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Street Address:</font></td>
<td>
<input type="text" name="BrokerStreetAddress" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">City:</font></td>
<td>
<input type="text" name="BrokerCity" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">State:</font></td>
<td>
<input type="text" name="BrokerState" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Zip:</font></td>
<td>
<input type="text" name="BrokerZip" size="10"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Phone:</font></td>
<td>

```

```

<input type="text" name="BrokerPhone" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Fax:</font></td>
<td>
<input type="text" name="BrokerFax" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">E-Mail Address:</font></td>
<td>
<input type="text" name="Email" size="30"></td>
</tr>
</TABLE>
<HR>
<table border="0" width="420">
<TR><TD COLSPAN="2" bgcolor="#FFCC66">
<FONT FACE="Arial" SIZE="2"><B>CLIENT INFORMATION</B></FONT>
</TD></TR>
<tr>
<td>
<font face="Arial" size="2">Name:</font></td>
<td>
<input type="text" name="Name" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Street Address: &nbsp; &nbsp; &nbsp;</font></td>
<td>
<input type="text" name="StreetAddress" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">City:</font></td>
<td>
<input type="text" name="City" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">State:</font></td>
<td>
<input type="text" name="State" size="30"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Zip:</font></td>
<td>
<input type="text" name="Zip" size="10"></td>
</tr>
<tr>
<td>
<font face="Arial" size="2">Phone:</font></td>
<td>
<input type="text" name="Phone" size="30"></td>
</tr>

```



```

</TD>
<TD WIDTH=45 ALIGN=RIGHT>$$picture$$</TD>
</TR></TABLE>
</td>
</tr>
<tr>
<td colspan="2" bgcolor="#CCCDAB"><font size="2"
face="Arial"><strong>Home Type:</strong>
$$HomeType$$</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#9CFADB"><font size="2"
face="Arial"><strong>Price Range:</strong> $$PriceLow$$ - $$PriceHigh$$
</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#CCCDAB"><font size="2" face="Arial"><strong>County:
</strong>$$County$$</font></td>
<td colspan="2" bgcolor="#CCCDAB"><font size="2" face="Arial"><strong>City: </strong>$$City$$</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#9CFADB"><font size="2"
face="Arial"><strong>Air Conditioned Square Footage:</strong> $$$SqfLow$$ - $$$SqfHigh$$
</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#CCCDAB"><font size="2"
face="Arial"><strong>Features:</strong> $$Features$$
</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#9CFADB"><font size="2"
face="Arial"><strong>Community Description:</strong> $$LifeStyle$$
Community<br>$$DescribeCommunity$$
</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#CCCDAB"><font size="2"
face="Arial"><strong>Driving Directions:</strong>
$$Directions$$
</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#9CFADB"><font size="2" face="Arial"><strong>Builder/Developer:</strong>
$$Builder$$</font></td>
<td colspan="2" bgcolor="#9CFADB"><font size="2" face="Arial"><strong>Phone:</strong> $$Phone$$</font></td>
</tr>
<tr>
<td colspan="2" bgcolor="#CCCDAB"><font size="2" face="Arial"><strong>E-mail:</strong></font>
<a href="mailto:$$Email$$"><font size="2" face="Arial" color="#0000ff">$$Email$$</font></a></td>
<td colspan="2" bgcolor="#CCCDAB"><font size="2" face="Arial"><strong>Website:</strong></font>
<a href="$$Web$$" target="_top"><font size="2" face="Arial" color="#0000ff">$$Web$$</font></a></td>
</tr>
<tr>
<td colspan="2" colspan="2">&nbsp;</td>
</tr>

```



```

src="http://www.brokerswelcome.com/images/buthead_builders.gif" width=200 height=30 border=0
alt="Realtors">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><A
href="http://www.brokerswelcome.com/register.html"></A></TD><td align="CENTER" valign="TOP"><a
href="http://www.brokerswelcome.com/logolink.html"></a></TD><td align="CENTER" valign="TOP"><a
href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>
<br>
<p>
<div align="center"></div><br>
<FORM name="ClassifiedForm" action="http://gator.twister.net/cgi-bin/db-add.cgi" method="POST">
<INPUT TYPE=HIDDEN NAME="FilePrefix" VALUE="anim">
<input type="Hidden" name="FormSubject" value="$$FormSubject$$">
<INPUT TYPE=HIDDEN NAME="RowID" VALUE="$$RowID$$">
<table border="0" align="center">
<tr>
<td><font face="Arial" size="2"><b>Enter Your<br>
Message:</b></font></td>
<td><font face="Arial" size="2"><textarea cols=30 rows=6 name="Body"
wrap="virtual">$$Body$$</textarea></font></td>
</tr>
<tr>
<td><font face="Arial" size="2"><b>E-mail:</b></font></td>
<td><font face="Arial" size="2"><input type="text" name="Email" size="30" value="$$Email$$"></font></td>
</tr>
<tr>
<td><font face="Arial" size="2"><b>UserName:</b></font></td>
<td><font face="Arial" size="2"><input type="text" name="UserName" size="30"
value="$$UserName$$"></font></td>
</tr>
<tr>
<td><font face="Arial" size="2"><b>Password:</b></font></td>
<td><font face="Arial" size="2"><input type="password" name="UserPassword" size="30"
value="$$UserPassword$$"></font></td>
</tr>
<tr>
<td></td>
<td align="center"><font face="Arial" size="2"><input type="submit" value="Enter">&nbsp;&nbsp;&nbsp;<input
type="Reset"></font></td>
</tr>
</table>
</form>
<BR>
<!-- BOTTOM LINKS -->
<CENTER><FONT FACE="Arial" SIZE="1">[ <a href="http://www.brokerswelcome.com/index.html">Home
Page</a> ] [ <a href="http://www.brokerswelcome.com/promos.html">Register to Receive Builder Incentives

```

```

and Promos</a> ] [ <A href="http://www.brokerswelcome.com/register.html">Register Your Clients with
Builders</A> ]</FONT><br>
<FONT FACE="Arial" SIZE="1">[ <a href="http://www.brokerswelcome.com/logolink.html">Get Your
"Brokers Welcome" Logo Link</a> ] [ <a href="http://www.brokerswelcome.com/mail.html">Mail Your
Broker Program</a> ]</FONT></CENTER>
<IMG SRC="http://www.brokerswelcome.com/images/rule_bott.gif" WIDTH=585 HEIGHT=20 VSPACE=5
BORDER=0 ALT="Rule">
</td>
</tr>
<!-- DISCLAIMER INFORMATION -->
<tr>
<TD></TD><td align="center"><DIV ALIGN="center"><TABLE><tr>
<td><A href="http://www.americanhomeguides.com/"></A></td>
<td><font face="arial" size="1">
Copyright &copy; 1999 American Home Guides, Inc.<br>
Phone: (954) 964-3717 Fax: (954) 964-2818<br>
E-mail to: <a
href="mailto:info@americanhomeguides.com">info@americanhomeguides.com</a><br></font></td>
</tr></table>
<BR>
<br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>
<br>
</td>
</tr>
</table></center>
</BODY>
</HTML>

```

17. **anim-failure.html**, this document is called when an error occurs while the builder fails in trying to create the animated creating card:

```

<HTML>
<HEAD>
<TITLE>Exclusive Florida Home Builder Broker Mail In Program</TITLE>
<meta name="description" content="Welcome to the Internet Post Office for Florida Home Builders. From here
you can send your Broker Program mailing to hundreds of Florida realtors who have registered with us, at the
click of your mouse.">
</HEAD>
<BODY background="http://www.brokerswelcome.com/images/bckgd.gif" bgcolor="White" link="Red"
vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG SRC="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5
BORDER=0 ALT="Rule">

```

```

<center><a href="http://www.brokerswelcome.com/index.html"></a></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>
<tr><td colspan="2" align="CENTER" valign="TOP"></TD><td colspan="2" align="CENTER" valign="TOP">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/register.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/logolink.html"></a></TD><td align="CENTER" valign="TOP"><a
  href="http://www.brokerswelcome.com/mail.html"></a></td></tr></table>
<blockquote><p align="center"><font face="Arial">Unable to add listing.</font></p></blockquote>
<BR>
<!-- BOTTOM LINKS -->
<CENTER><FONT FACE="Arial" SIZE="1">[ <a href="http://www.brokerswelcome.com/index.html">Home
  Page</a> ] [ <a href="http://www.brokerswelcome.com/promos.html">Register to Receive Builder Incentives
  and Promos</a> ] [ <a href="http://www.brokerswelcome.com/register.html">Register Your Clients with
  Builders</a> ]</FONT><br>
<FONT FACE="Arial" SIZE="1">[ <a href="http://www.brokerswelcome.com/logolink.html">Get Your
  "Brokers Welcome" Logo Link</a> ] [ <a href="http://www.brokerswelcome.com/mail.html">Mail Your
  Broker Program</a> ]</FONT></CENTER>
<IMG SRC="http://www.brokerswelcome.com/images/rule_bott.gif" WIDTH=585 HEIGHT=20 VSPACE=5
  BORDER=0 ALT="Rule">
</td>
</tr>
<!-- DISCLAIMER INFORMATION -->
<tr>
<TD></TD><td align="center"><DIV ALIGN="center"><TABLE><tr>
<td><a href="http://www.americanhomeguides.com/"></A></td>
<td><font face="arial" size="1">
Copyright &copy; 1999 American Home Guides, Inc.<br>
Phone: (954) 964-3717 Fax: (954) 964-2818<br>
E-mail to: <a
  href="mailto:info@americanhomeguides.com">info@americanhomeguides.com</a><br></font></td>
</tr></table>
<BR>
<br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>

```

[illegible]

```
<br><br><br><br><br>
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="center"><DIV ALIGN="center"><TABLE><tr>
<td><A href="http://www.americanhomeguides.com/"></A></td>
<td><font face="arial" size="1">
Copyright &copy; 1999 American Home Guides, Inc.<br>
Phone: (954) 964-3717 Fax: (954) 964-2818<br>
E-mail to: <a
href="mailto:info@americanhomeguides.com">info@americanhomeguides.com</a><br></font></td>
</tr></table>
<BR>
<br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>
<br>
</td>
</tr>
</table></center>
</BODY>
</HTML>
```

```
<HTML>  
<HEAD>  
<TITLE>Exclusive Florida Home Builder Broker Mail In Program</TITLE>  
<meta name="description" content="Welcome to the Internet Post Office for Florida Home Builders. From here  
you can send your Broker Program mailing to hundreds of Florida realtors who have registered with us, at the  
click of your mouse.">  
</HEAD>  
<BODY bgcolor="#FFFFFF" link="Red" vlink="Maroon" alink="Red">  
<center><table width="585" border="0" cellspacing="0" cellpadding="0">  
<!-- GUTTER -->  
<tr>  
<td width="570">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~</td>  
</tr>  
</table>
```

UserName=All&UserURL=www.floridaguide.com&UserPassword=fguide2100&UserFullName=American%20Home%20Guides
&FormSubject=Grand%20Opening%20Invitation&RowID=935184218&Email=blynn@gate.net&Body=You%20are%20invited%20to%20celebrate%20the%20grand%20opening%20of%20our%20new%20models%20at%20the%20Vinings%20

6pm%20Thursday%20September%202023,%201999<

BR>
Cocktails%20and%20fine%20food%20will%20be%20served.

RSVP%20by%20Email%20by%20September%2015th%20or%20call%20954-964-2818

The%20Vinings
123%20Amber%20lane
Hollywood,%20FL%2033020

21. anim-listings.b, this is the buffer for the database which contains the same data as anim-listings.lis:

```
UserName=All&UserURL=www.floridaguide.com&UserPassword=fguide2100&UserFullName=American%20Home%20Guides<BR>&FormSubject=Grand%20Opening%20Invitation&RowID=935184218&Email=blynn@gate.net&Body=You%20are%20invited%20to%20celebrate%20the%20grand%20opening%20of%20our%20new%20models%20at%20the%20Vinings%20<BR><BR>6pm%20Thursday%20September%2023,%201999<BR><BR>Cocktails%20and%20fine%20food%20will%20be%20served.<BR><BR>RSVP%20by%20Email%20by%20September%2015th%20or%20call%20954-964-2818<BR><BR>The%20Vinings<BR>123%20Amber%20lane<BR>Hollywood,%20FL%2033020
```

22. anim-success.html, this is the file that is called upon when the builder successfully created his animated greeting card:

```
<HTML>
<HEAD>
<TITLE>Exclusive Florida Home Builder Broker Mail In Program</TITLE>
<meta name="description" content="Welcome to the Internet Post Office for Florida Home Builders. From here you can send your Broker Program mailing to hundreds of Florida realtors who have registered with us, at the click of your mouse.">
</HEAD>
<BODY background="http://www.brokerswelcome.com/images/bckgd.gif" bgcolor="White" link="Red" vlink="Maroon" alink="Red">
<center><table width="585" border="0" cellspacing="0" cellpadding="0">
<!-- GUTTER -->
<tr><td WIDTH="15">

</td>
<!-- TEXT FIELD -->
<td width="570">
<IMG SRC="http://www.brokerswelcome.com/images/rule_top.gif" WIDTH=585 HEIGHT=20 VSPACE=5 BORDER=0 ALT="Rule">
<center><a href="http://www.brokerswelcome.com/index.html"></a></center>
<!-- BUTTONS -->
<br>
<table width="570" border="0" cellspacing="0" cellpadding="0" nowrap>
<tr><td colspan="2" align="CENTER" valign="TOP"></TD><td colspan="2" align="CENTER" valign="TOP">
</TD></TR>
<tr><td align="CENTER" valign="TOP">
<a href="http://www.brokerswelcome.com/promos.html"></a></TD><td align="CENTER" valign="TOP"><a href="http://www.brokerswelcome.com/register.html"></a></TD><td align="CENTER" valign="TOP"><a href="http://www.brokerswelcome.com/logolink.html"><br>
<font size="1" face="Arial">Equal Housing Opportunity<br>
Void in states where prohibited by law.</font><br>
<br>
</td>
</tr>
</table></center>
</BODY>
</HTML>

```

24. **anim-template.html**, this is the template that is used to view the animated greeting card:

```

<table width="600" border="0">
<tr>
<td align="center" valign="top"></td>
<td align="center"><table width="100%" height="281" border="0" cellpadding="0" cellspacing="0">
<tr>
<td height="85%"><font face="Arial" size="3" color="#0000FF">$$Body$$<br><br></font></td>
</tr>
<tr>
<td valign="bottom" height="15%"><font face="Arial" size="3"><a
  href="http://$$UserURL$$">$$UserFullName$$</a></font>
<font face="Arial" size="3"><a href="mailto:$$Email$$">$$Email$$</a></font><br></td>
</tr>
</table></td>
</tr>
</table>
<p>
<embed src="http://www.brokerswelcome.com/GrandOpening.mid" width=2 height=0 autostart=true loop=true
  hidden="True">

```

25. **anim-template2.html**, this is the template that is used to mail out the animated greeting card:

```

<table width="600" border="0">
<tr>
<td align="center" valign="top"></td>
<td align="center"><table width="100%" height="281" border="0" cellpadding="0" cellspacing="0">
<tr>
<td height="85%"><font face="Arial" size="3" color="#0000FF">$$Body$$<br><br></font></td>
</tr>
<tr>
<td valign="bottom" height="15%"><font face="Arial" size="3"><a
  href="http://$$UserURL$$">$$UserFullName$$</a></font>
<font face="Arial" size="3"><a href="mailto:$$Email$$">$$Email$$</a></font><br></td>
</tr>
</table></td>
</tr>
</table>
<tr>
<td align="center" colspan="2">
<form action="/cgi-bin/db-list.cgi" method="post">
<input type="Hidden" name="FilePrefix" value="em">
<INPUT type="Hidden" name="ToMailList" value="True">
<input type="Hidden" name="FormFrom" value="$$Email$$">

```



```
<input type="Hidden" name="FormSubject" value="$$FormSubject$$">
<input type="Hidden" name="UserName" value="$$UserName$$">
<input type="Hidden" name="UserPassword" value="$$UserPassword$$">
<input type="Hidden" name="FormText" value="You have been invited to a Grand Opening. Please click the
address below to get your invitation.
```

```
http://gator.twister.net/cgi-bin/db-list.cgi?FilePrefix=anim&RowID=$$RowID$$
```

```
Thank you,
```

```
$$UserFullName$$">
```

```
<table border="0" width="420">
```

```
<tr>
```

```
<td width="100%" align="center" colspan="2"><br><br><br>
```

```
<font face="Arial" size="3"><b>Send Your Card</b></font></td>
```

```
</tr>
```

```
<tr>
```

```
<td width="30%" valign="top">
```

```
<font face="Arial" size="2">Select Region:</font></td>
```

```
<td width="70%"><font face="Arial" size="1" color="#000000"><select name="Region" size="1">
```

```
<option value=""> </option>
```

```
<option value="Northeast: Jacksonville, St. Augustine, Gainesville">Northeast: Jacksonville, St. Augustine,
Gainesville </option>
```

```
<option value="Northwest: Tallahassee, Pensacola, Panama City">Northwest: Tallahassee, Pensacola, Panama
City </option>
```

```
<option value="East Central: Melbourne, Orlando, Daytona">East Central: Melbourne, Orlando, Daytona
</option>
```

```
<option value="South Central: Stuart, Port St. Lucie, Ft. Pierce, Vero Beach, Sebring, Lakeland">South Central:
Stuart, Port St. Lucie, Ft. Pierce, Vero Beach, Sebring, Lakeland </option>
```

```
<option value="West Central: Tampa, St. Petersburg, Ocala, Brooksville">West Central: Tampa, St. Petersburg,
Ocala, Brooksville </option>
```

```
<option value="Southwest: Naples, Ft. Myers, Sarasota, Punta Gorda">Southwest: Naples, Ft. Myers, Sarasota,
Punta Gorda </option>
```

```
<option value="Palm Beach: Boca Raton, Delray Beach, Boynton Beach, Palm Beach, Jupiter">Palm Beach:
Boca Raton, Delray Beach, Boynton Beach, Palm Beach, Jupiter </option>
```

```
<option value="Broward: Hollywood, Pembroke Pines, Ft. Lauderdale, Coral Springs">Broward: Hollywood,
Pembroke Pines, Ft. Lauderdale, Coral Springs </option>
```

```
<option value="Dade/Monroe: Miami, Miami Beach, Kendall, Florida Keys">Dade/Monroe: Miami, Miami
Beach, Kendall, Florida Keys </option>
```

```
</select></font>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
<input type="submit" value="Send Your Card">
```

```
</form></td>
```

```
</tr>
```

```
</table>
```

```
<p>
```

Section II

Usage instructions for both builders and Real Estate professions are maintained at <http://www.brokerswelcome.com>. From there, four transactions can occur.

1. Real Estate professionals can add their names and contact information to the mailing list by going to <http://www.brokerswelcome.com/promos.html>. There they have an

option of either joining for free which does not include a link to their website or email, or purchasing an enhanced listing which includes a link to their website and email. The price is only \$29 for 6 months. They enter in their name, company, physical address, region within their state, email address and phone number (an example of the form is located at http://www.brokerswelcome.com/state_free.html) and press submit. The information is then stored in the database (brk-listings.lis). This form uses the db-add.cgi program as well as the brk-header.html, brk-footer.html, (brk-success.html or brk-failure.html), brk-listings.lis and brk-pass.lis

2. Real Estate Professionals can register their clients with homebuilders. To accomplish this, they must go to <http://www.brokerswelcome.com/register.html>, fill out the form with their search criteria (city, county, price range and type of home) and press the "Search" button. This form uses db-list.cgi to query the com-listings.lis database, which is the database, where the new home community listings are stored. Once the results have been obtained from the database, it is then displayed onto the screen using reg-header.html as the header, reg-com.tmpl as the template and reg-footer.html as the footer. The results display descriptive listings of communities where the builders have agreed to permit online client registration. From these listings, the Real Estate Professional can click onto the builder's web site for more information. Should the Real Estate Professional decide to register his client at the community, he clicks on to "Click here to register with this community" which appears above of every listing. Once he clicks on this link, the db-form.cgi program using fields that were passed to it by the db-list.cgi program as well as with reg-next.html, which is the template, generates the next page. The Real Estate professional then enters in his contact information as well as his clients contact information and clicks the submit button. Once this is done, the LYNQS_bw_email.pl program then emails this information to the builder, to the Real Estate professional and to Barry Lynn (to monitor the activity of the program as well as to have documentation in the event of a dispute) and displays a thank you note using brokers-welcome_header.html as the header and brokers-welcome_footer.html as the footer.
3. Builders can obtain the HTML coding for the "Brokers Welcome" logo link, which needs to be added to their web sites at <http://www.brokerswelcome.com/logolink.html>. This logo link is designed to induce Real Estate professionals visiting the builders' web site to click it and add themselves to the database.
4. Builders can send email to the Real Estate professionals who have added themselves to the database (brk-listings.lis). They have two choices: A text message which is located at <http://www.brokerswelcome.com/mail.html> or an animated "greeting card" with music which is located at <http://www.brokerswelcome.com/mail-anim.html>.

When sending a text message the builder enters in the desired region he wants to email, the subject of the email, the email message, his email address, the username, the password and presses the "Mail To Realtors" button. The db-list.cgi performs a query for all the realtors that are located in that region and begins to send the email message that the builder entered to each Realtor. The screen will show each email address that it was sent to as well as which email addresses had errors. The email message will have a note at the bottom that reads "To be removed from this list please

click on the link below: <http://gator.twister.net/cgi-bin/db-delete.cgi?FilePrefix=brk?RowID=#####>".

To send an animated greeting card the builder enters in the email message, his email address, his username, his password and he then must click the "submit" button to continue. The db-add.cgi program then adds this information to the anim-listings.lis database and displays the anim-succes.html page, which prompts the builder to enter in his username and password again and press the "Enter" button. Once this is done, the db-list.cgi program then displays the builders greeting using the anim-header2.html as the header and the anim-template2.html as the template. The builder also has the option of deleting the card by click onto "Click to Delete" which appears above the card, editing the card by clicking onto "Click to Edit" which appears above the listing as well, or he could select a region from the drop down box and press "Send Your Card" to send his greeting card to all the realtors in that region using the db-list.cgi program to perform a query on the brk-listings.lis database. The email will have a link to the animated greeting card as well as a note at the bottom that reads "To be removed from this list please click on the link below: <http://gator.twister.net/cgi-bin/db-delete.cgi?FilePrefix=brk?RowID=#####>".

The invention described herein extends to all such modifications and variations as will be apparent to the reader skilled in the art, and also extends to combinations and subcombinations of the features of this description and the accompanying Figs and Example. Although preferred embodiments of the methods and systems of the present invention have been illustrated in the accompanying Figs. and described in the foregoing Detailed Description and Example, it will be understood that the invention is not limited to the embodiments disclosed, but is capable of numerous rearrangements, modifications and substitutions without departing from the spirit of the invention as set forth and defined by the following claims.

Having described my invention, I claim: